A Logic Framework of Normative-based Contract Management

Miao Wang and Guido Governatori
School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, Queensland, 4072, Australia

1 Introduction

FIDIC is the acronym of International Federation of Consulting Engineers (the abbreviation represents the French version of the name). Founded in 1913 FIDIC has now become an international federation of national associations of consulting Engineers including 67 Member Associations from all parts of the world. The series of standard contract forms stipulated by FIDIC has become international standards in project management, including civil, mechanical, electricity, medicine engineering projects (for a comprehensive survey, please check www.fidic.org for more details).

In this paper we combine the defeasible logic of institutionalised agency [3, 1, 4] with the defeasible logic of temporalised normative positions [5]. The resulting logic offers a computationally oriented (non-monotonic) formal framework for the representation of normative-based contract management. In particular the logic offers facilities to represent abstract agency (implementing a "brings it about" modal operator), counts-as conditionals, and normative conditionals for obligations, permissions, prohibitions and violations. The temporal extension allows us to reason with deadlines as well as the initiation and termination of normative positions.

The focus of the paper is to demonstrate that the proposed logical framework is capable to represent and to implement real life contract management standards. In particular we propose a logical encoding of one clause of the FIDIC regulation: Clause 67 about the rules of the disputes and the corresponding procedures of solving the disputes. The encoding provides an executable specification of the clause, and we test it against a simple scenario.

2 Logical Framework

The logical framework is based on defeasible logic. Defeasible Logic is a rule based non-monotonic formalism. A rule in Defeasible Logic corresponds to a relationships between a set of premises (literals) and a conclusion (literal). In

Paper presented at *Formal Methods in Electronic Commerce Workshop 2007*, Stanford University, Palo Alto, 4 June 2007

Defeasible Logic we have three types of rules: strict rules, which are rules in the classical sense, $a_1, \ldots, a_n \rightarrow b$, meaning that b is a definite/indisputable conclusion of $a_1, \ldots, a_n$ when these are indisputable; defeasible rules, $a_1, \ldots, a_n \Rightarrow b$, meaning that usually b is a conclusion of $a_1, \ldots, a_n$ unless there is some stronger evidence against it. Finally we have defeaters. Defeaters are the weakest relationships between a set of premises and a conclusions, a defeater like $a_1, \ldots, a_n \rightsquigarrow b$, is can be understood as a reason to prevent the derivation of $\neg b$ from $a_1, \ldots, a_n$, but not to support the conclusion of b from the same set of premises.

Beside the above classification of rule based on their strength, we have a second classification based on the type of conclusion (mode) we have: in this paper we consider "counts-as" rules (which will also be used as the basic inferential mechanism of the logic), "results-in" rules, and "obligation" rules. Defeaters have a special meaning for obligation rules, instead of creating a new obligation, they terminate an existing one.

To capture the temporal dimension, we introduce (1) temporalised literals, e.g., $a : t$ meaning that a holds at time t where t is a timestamp, and (2) for each rule we specify whether the conclusion is transient, i.e., it holds only for a particular instant, or permanent, i.e., it continues to hold until a terminating event occurs.

Based on the above discussion we can have rules like:

$$a_1 : t_1, \ldots a_n : t_n \Rightarrow^{per}_{C} b : t \tag{1}$$

$$a_1 : t_1, \ldots a_n : t_n \Rightarrow^{tr}_{E_j} b : t \tag{2}$$

$$a_1 : t_1, \ldots a_n : t_n \Rightarrow^{per}_{O_j} b : t \tag{3}$$

$$a_1 : t_1, \ldots a_n : t_n \rightsquigarrow^{per}_{O_j} b : t \tag{4}$$

Here (1), a counts-as rule, states that in the context of the FIDIC contract under analysis if a_i holds at time t_i (for all i, $1 \leq i \leq n$), then we are allowed to assert that b holds at time t, and, since the rule is labelled as persistent, the effect b will hold after t until some event eventually terminates the validity of b. For (2), a results-in rule, the meaning is that if a_i holds at time t_i (for all i, $1 \leq i \leq n$), then we are allowed to assert that agent j does b at time t (or j brings is about that b at time t, $E_j b : t$). However, this is a transient rule, so we cannot assert that j achieved b at a time t', $t' > t$. Rule (3) is an obligation rule that specifies that agent j has the obligation b at time t ($O_j b : t$), when the conditions a_i to a_n are satisfied at the appropriate times, and that the obligation persists, until a discharging event occurs. Finally, (4) terminates the obligation on $\neg b$ (prohibition of b) at time t. For further details see [4, 5, 3, 2].

3 Encoding Clause 67 in Defeasible Logic

Clause 67 of FIDIC regulation sets down condition to begin dispute and to handle them. The encoding for this clause has two parts: in the first part we

put down some general rules about disputes and then we have specific parts for each of the four articles of the clause. We include only a few rules to illustrate the main features of our framework.

In the first stage, the *Employer* or the *Contractor* should send the dispute to the *Engineer* in writing. Then the *Engineer* has at most 12 weeks to consider this referred dispute and draw his decision towards it. In this period both parties of the *Employer* and the *Contractor* should go on to obey the *Engineer*'s original/old decision/certificate about the dispute matter, which has already encoded in $r4.1_{general}$ and $r4.2_{general}$ (not shown here).

Several actions needed to be set here.

Action $A1$ (the *Employer* or the *Contractor*) to refer a dispute to the Engineer.

Action $A2$ (the *Engineer*) to draw a notice of decision.

$$
\begin{aligned}
R1.1.1 &: E_{em}A1 : t1 \Rightarrow^{tr}_{C} BeginOfDispute : t1 \\
R1.1.2 &: E_{co}A1 : t1 \Rightarrow^{tr}_{C} BeginOfDispute : t1 \\
R1.2 &: BeginOfDispute : t1 \Rightarrow^{per}_{O} E_{en}A2 : [t1, t1 + 84] \\
R1.3.1 &: (E_{en}A2 : t2), (t1 < t2 < t1 + 84) \Rightarrow^{tr}_{C} NoticeOfDecision : t2 \\
R1.3.2 &: (E_{en}A2 : t2), (t1 < t2 < t1 + 84) \rightsquigarrow^{per}_{O} \neg E_{en}A2 : t2 + 1
\end{aligned}
$$

The interval representation, i.e., $[t1, t1 + 84]$ in rule $R1.2$ is just syntactic sugar for a second rule that terminate the obligation at time $t1 + 84$. Please pay attention to $R1.3.2$. It means if the Engineer fulfils his obligation successfully at time point t then she is discharged from that obligation form time point $t+1$.

From the deontic rule $R1.2$ we get the corresponding violation rule $R1.2.1$. If the Engineer does not draw a conclusion in 84 days, it is a violation to the general rule $R1.2_{general}$ as described before.

$$
\begin{aligned}
R1.2.1 : (BeginOfDispute : t1), (O_{en}E_{en}A2 : t1 + 84), \\
(E_{en}\neg A2 : t1 + 84) \Rightarrow_{C} Vl_{en}((em), R1.2_{general}) : t1 + 84
\end{aligned}
$$

This violation would trigger the general rule $R3.2_{general}$, so the Engineer cannot get payment at time point $t1 + 84$ for the related dispute solution work.

References

[1] Guido Governatori and Antonino Rotolo. A computational framework for non-monotonic agency, institutionalised power and multi-agent systems. In Daniéle Bourcier, editor, *Legal Knowledge and Information Systems*, number 106 in Frontieres in Artificial Intelligence and Applications, pages 151–152, Amsterdam, 2003. IOS Press.

[2] Guido Governatori and Antonino Rotolo. A defeasible logic of institutional agency. In Gerhard Brewka and Pavlos Peppas, editors, *NRAC'03*, pages 97–104, Acapulco, Mexico, 10-11 August 2003. IJCAI.

[3] Guido Governatori and Antonino Rotolo. Defeasible logic: Agency, intention and obligation. In Alessio Lomuscio and Donald Nute, editors, *Deontic Logic in Computer Science*, number 3065 in LNAI, pages 114–128, Berlin, 2004. Springer-Verlag.

[4] Guido Governatori and Antonino Rotolo. A computational framework for institutional agency. *Artificial Intelligence and Law*, 2007.

[5] Guido Governatori, Antonino Rotolo, and Giovanni Sartor. Temporalised normative positions in defeasible logic. In Anne Gardner, editor, *10th International Conference on Artificial Intelligence and Law (ICAIL05)*, pages 25–34. ACM Press, June 6–11 2005.